The Mad Hatter Meets the MCSE

By Paul Murphy – September 2003

Introduction

Taking a 100% Microsoft client-server systems environment and simply adding Linux based desktops to the mix is a bit like dropping a mongoose into a snake pit -- short term savings on licensing and hardware will be eaten up by increased support costs and decreased operational stability.

One alternative is to adopt Unix throughout the enterprise, earning savings in the 40% range while improving reliability and performance by combining centralized processing with simplified networking, desktop smart displays, and enhanced user control over corporate data processing.

In the movie *Raiders of the Lost Ark* Indiana Jones is confronted by an obviously highly skilled and aggressive local sword fighter who clearly expects to hack up our hero after a glorious, and ritualistic, hand to hand battle. Unfortunately Dr. Jones is too harried to play, and shoots him dead at twenty paces in a classic demonstration of what happens when a clearly superior technology appears on the scene.

From the perspective of those who pay for or use corporate computing, Unix can offer superior technology but the real life transition won't be as simple or dramatic as the movie metaphor. To the MCSE, the Lintel desktop isn't a revolver; it's a cheaper sword. Their natural, and probably only, response to the Linux desktop will therefore be to integrate it into the client-server architecture, turning it into a kind of second rate Windows clone suffering from all the instability, and support needs, of the Microsoft suites it replaces.

Part One: Alice meets the Mad Hatter

`Yes, that's it,' said the Hatter with a sigh: `it's always tea-time, and we've no time to wash the things between whiles.'

Sun's "Project Mad Hatter" desktop solution is the latest, and highest profile, entry to the corporate Linux desktop market. Although the software will also be available on Solaris, its fundamental purpose is to turn a PC running Linux into a near clone of one running Microsoft Office on some Microsoft brand Windows operating system. The package builds on existing Sun and open source technology such as StarOffice and Gnome, with the aim to improve setup and administration costs, ease of use, and security. Mad Hatter's basic selling propositions are:

1. The interface and key software are sufficiently Microsoft Windows-like to be non threatening to most users;
2. License costs are significantly lower;
3. The transition can be staged with those who need the ostensibly more powerful, or at least more traditional, Microsoft desktop unaffected by change among lesser users;
4. Hardware churn is minimized --typically extending the useful life of a desktop PC from around 24 months to something closer to the physical life of the hardware;
5. Sun offers legal protection covering desktop components like Star office against third party intellectual property claims that aren't available to companies sourcing their Linux desktops from the IBM/SuSe partnership or other players.

This marketing approach to selling the Linux desktop appears to envisage a gradual take up as companies experiment with the Mad Hatter desktop on existing PCs and eventually roll them out to the 80% or so of their users for whom full Microsoft compatibility is considered non critical.

In contrast the IBM/SuSe approach to selling the Linux desktop is geared toward big bang transitions with organizations like German metropolitan governments expected to throw out tens of thousands of Microsoft licenses and hire IBM to implement Linux instead.

Both sales pitches, however, incorporate some elements of the "build on what you know" strategy in which the customer's executive is encouraged to see the change to Lintel (Linux on Intel) as little more than the substitution of "good enough" applications running on a better OS at lower cost.

Both pitches have their appeal; together they should lead to significant acceleration in the deployment of Linux desktops in government and business organizations worldwide. It is this potential for explosive growth which raises the question around which we're gathered here today: what does the arrival of Lintel mean in terms of the organization's need for the Microsoft gurus who currently support the organization's Wintel ([Microsoft] Windows on Intel) desktops?

Or, to phrase the question from a marginally different perspective: how does the typical MCSE skill set map to what will be needed to cope with an environment in which perhaps 20% of the servers and 80% of the desktops run Linux while the remainder continue to run Microsoft suites?

The MCSE exam list suggests that the typical MCSE offers technical expertise with respect to:

1. Managing and maintaining a Microsoft Windows Server 2003 environment
2. Implementing, managing, and maintaining a Microsoft Windows Server 2003 network infrastructure
3. Planning, implementing, and maintaining a Microsoft Windows Server 2003 Active Directory infrastructure.
4. Installing, configuring, and administering Microsoft Windows XP Professional
5. Designing security for a Microsoft Windows Server 2003 network
6. Installing, configuring, and administering Microsoft SQL Server 2000 Enterprise Edition

Obviously none of the memorized click sequences typical of the Microsoft re-boot and re-install cycle will apply, but what about the more technical reflexes and management certainties that grow from working with Microsoft tools and ideas? Will any of these retain value for organisations converting to a largely Linux desktop environment?

The short, cynical, but probably fair, answer is that application of the expertise created by MCSE training and experience can be expected to increase the cost of use for enterprise Linux desktops while decreasing their effectiveness.

In computing, as elsewhere, expertise decays in the presence of technical change, leaving only out dated reflexes and position in its place. We saw this when the first Vax based Unix servers were reviled by data processing professionals for not offering solutions, like virtualization and partitioning, to process management and cost problems Unix doesn't have. We're seeing it again today as Wintel gurus load one application per Linux server, use rackmounts to simulate SMP, and demand Wintel style cluster management and load balancing software while ignoring both the capabilities of the Linux kernel and the architectural implications of Unix licensing.

MCSE Skill Survival

	MCSE Key Skill Set	Portable to Linux?
1	Performing an Attended Installation of Windows XP Professional	No
2	Performing an Unattended Installation of Windows XP Professional	No
3	Configuring and Troubleshooting the Desktop Environment	No
4	Managing Windows XP Hardware	Partial
5	Configure and Manage Windows XP Devices	No
6	Configuring Disk Drives and Volumes	No
7	Configure and Manage Windows XP Printing and Faxing	No
8	Resource Administration	No
9	Microsoft Networking	No
10	Internet and Remote Networking	Partial
11	Configuring, Managing, and Troubleshooting Security	Partial
12	Monitoring and Optimizing System Performance and Reliability	No

So what can we expect when Linux desktops are added to the corporate support mix? There's no reason to think that the same failure to adapt technical behavior to the technology won't affect corporate desktop Lintel deployments whether implemented gradually as supposed by Sun or holus bolus as suggested by IBM.

Consider, for example, how core MCSE skill sets like "Managing and maintaining a Microsoft Windows Server 2003 environment" or "Planning, implementing, and maintaining a Microsoft Windows Server 2003 Active Directory infrastructure" will be affected by a decision to convert up to 80% of an organization's desktops to Lintel. With the exception of security management, essentially all of the practical skills associated with those functions will be invalidated. DHCP, WINS, SMB networking, Processor Affinity Management, Domain Administration, Registry hacking, and so on, are all technologies and ideas out of place in a well run Unix environment, though some pollution has crept in.

A few bits of knowledge will contribute to the MCSE's ability to work in the new environment. For example, much of Microsoft's TCP/IP use (known as external and remote networking in the Wintel world) is applicable to Unix because, of course, it is fundamentally from Unix. Obviously, most of the specific skills associated with this will have to be abandoned but, much more importantly, many of the key Wintel compromises developed to allow SMB networked PCs internet access -- including ideas about proxies, gateways, cryptology clients, and so on -- can be, *and therefore will be,* perpetuated in the new environment despite having no natural role there.

It is reflexive management and technical certainties like these which provide the most obvious vectors for unnecessary cost and complexity but the differences go far beyond such relatively superficial issues. Consider, for example, what the habits of thought created by extensive use of Microsoft Word or Excel either as, or within, application clients mean for people suddenly left without VB as an external integration engine.

It's true, of course, that these technologies should never have existed and that they were forced onto the Microsoft Windows community as work-arounds for the limitations of the Microsoft operating systems involved, but none of that means they'll give them up. Instead people will try to replicate what they know in the new environment and that effort will have pyramiding negative effects on costs, system stability, and overall productivity.

`Not the same thing a bit!' said the Hatter. `You might just as well say that "I see what I eat" is the same thing as "I eat what I see"!'

It isn't any harder to send a Linux user a piece of email with a malicious attachment than it is to send one to Wintel user. But while the Wintel user can execute an attachment with a trivial mouse click, the Linux user generally has to perform a few non-obvious actions before that attachment can become executable and therefore dangerous.

However, buffer overflow and related attacks take advantage of software errors to gain privileged access to systems and run an attacker's code. As a result we can reasonably expect the security industry to sell its tools and ideas to Linux desktop users and thus allow this bit of MCSE expertise to perpetuate itself.

As an aside, many RISC implementations support a non-executable stack that can nullify the threat of certain stack buffer overflow attacks (though it is not a silver bullet for all buffer overflow exploits). This paper discusses some of the differences between SPARC and x86 when it comes to buffer overflows. Efforts have been made to include support for non-executable stacks in Linux/x86 through patches available from, for instance, the Openwall Project.

If the Lintel for Wintel swap offered tremendous reductions in IT manpower, companies could cope with this kind of human intellectual inertia by hiring a few Unix people, letting them take over, and then laying off the MCSE hordes now dominant in IT.

Those reductions in staffing won't happen in part because existing IT groups will loudly protest their competence to handle the new software even while asserting the superiority of their ideas to Unix tradition and design; and in part because companies unwilling to outsource desktop support won't have alternatives; but mainly because the change doesn't affect the fundamental client-server architecture.

It is the impracticality of client-server, not the specifics of whether you use Wintel or Lintel to implement it, which drives up costs. It is quite true that Unix, whether presented as Linux, BSD, or Solaris is markedly better than the current 17 or so major Microsoft Windows brand OS variants, but when we've seen this kind of quantum leap in desktop OS quality before, the effects weren't exactly positive. For example, when Microsoft abandoned the last systems based on QDOS (Windows 9X/ME) in favor of systems based on VMS (NT 3.51/2000) the quality improved dramatically -- perhaps as much as the leap from Wintel to Lintel-- but not too many people claim they experienced cost decreases because of it.

Organizationally the impracticalities of the change are over-whelming. It is simply not possible to hire enough people who understand Linux to really attack support costs; in part because there aren't many, but more importantly because they won't work on help desks or as isolates surrounded and supervised by Wintel people.

So what are the alternatives to postponing the change or simply paying the costs as the MCSE community remakes Lintel in Wintel's image?

IBM's answer is clear. Outsource support to them, and they'll put their people in on your ground floor to help you through the transition.

Sun's Mad Hatter desktop provides only a small part of their strategic response to this challenge. If Lintel is where you see your future you can implement Mad Hatter easily enough --just load up Linux and the desktop suite in place of some Microsoft OS and Office and start learning how to run it.

The second part of Sun's answer is much more revolutionary. In Part Two, below, I'll introduce the Unix Business Architecture (UBA) before going on, in Part Three, to cost comparisons.

Part Two: Alice Finds a Golden Key

`It's a friend of mine--a Cheshire Cat,' said Alice: `allow me to introduce it.'

There are tens of thousands of organisations whose enjoyment of the Wintel desktop revolution has been somewhat tempered lately by high costs, security failures, and a sharpening executive focus on reliability issues.

One of the more important causes for people's willingness to voice their increasing unhappiness with Wintel is the growing popular awareness of Linux as an alternative. To that end the IBM/SuSe partnership offers extensive support for complete organizational IT make-overs, most of the big consultancies are eyeing Lintel as a potential successor to SAP in terms of revenue generation, and Sun is about to release an easily accessible application bundle, code-named Mad Hatter, which basically turns a Wintel PC into a Lintel PC.

Imagine, therefore, that you are one of the affected CIOs, you have several thousand Wintel desktops, costs are continuing to escalate, and you can't go to the executive washroom anymore without somebody accosting you about lost records, obsolete gear, or the joys of going with Linux, What do you do?

There are two fundamental choices:

1. Explore change opportunities; or,
2. Buckle down to making Wintel work.

There are three reasonable alternatives to continuing with a 100% Microsoft architecture:

1. The Lintel option involves converting up to 100% of your PC gear to running Linux, and Linux applications;
2. The Macintosh option involves converting desktops to MacOS X and your servers to whatever combination of Unix machines and software meets your needs; typically a mix of Apple's X-servers and high end RISC machines like Sun's SPARC servers.
3. The Unix Business Architecture option requires radical change at both the technology and management levels but promises a highly productive, hassle free, and very low cost computing environment.

What's a UBA?

The Unix Business Architecture centralises processing while decentralising control. Technically it combines big servers in the data center with smart displays on desktops and formal organisational support for Unix (Linux, BSD, Solaris, or MacOS X) at home or where high end desktop power is warranted.

Smart displays are not thin clients. The Microsoft thin client derives from Sun's 87/88 ELC diskless workstations and functions by downloading applications, or application clients, from a server and then running them locally.

Under the UBA all applications run on the server with the smart display handling only user interaction.

The 1998 IBM Redbook: IBM Network Station - RS/6000 notebook" by Laurent Kahn and Akihiko Tanishita. provides an excellent, if now somewhat dated, introduction to smart display technology covering set-up, operations, benefits, and typical business deployments.

Except for the use of Mac laptops within the Unix architecture costed below -- a choice based on the particular needs of sales people combined with operational cost issues -- the Macintosh option is not considered further in this article. It is, however, is far better than most people now seem to realize. There are three key reasons for this:

1. MacOS X running on a PowerPC CPU is tougher to crack compared to the security and processing continuity issues affecting Wintel. In particular, the underlying BSD operating system has standard Unix security while the platform is collectively a smaller target for the attention of worms and viruses.
2. The combination of Unix with the MacOS X GUI and PDF based graphics puts Apple well ahead of Microsoft in terms of desktop user interface design and OS implementation; and,
3. Apple's latest foray into enterprise products, the X-serve and X-serve RAID data array (2.5TB on 2GB/S fibre channels for less than $11K), coupled with standard Unix networking, security, and shared filesystem access now make it possible to fully implement an uncompromised departmental architecture entirely with Apple products.

Our focus, however, will be on the choice between Lintel and the Unix Business Architecture, or UBA, for an organization now deeply committed to Wintel.

According to a very interesting study: *Managing Your IT Total Cost of Ownership* by Julie Smith David, David Schuff, and Robert St. Louis in Communications of the ACM. (January 2002, pp101ff), Wintel costs are inversely correlated with centralization. In other words, the total cost of ownership for a Wintel infrastructure is minimized with maximum server centralization and desktop lockdown.

Since any good IT manager facing increased cost and user dissatisfaction in the Wintel environment will adopt Wintel best practices before considering more significant change, we can assume that most organisations now considering Lintel will already have adopted the maximum centralization approach to Wintel management. They will, in other words, have replicated the old IBM mainframe structure with centralized control, centralized servers, desktop fat clients, with or without Citrix and server based processing, in place of 32XX terminals.

In most cases the first Linux deployments attempted by Wintel architecture organisations will have been limited to external server functions like web services and email with usage gradually spreading to include file and print support for workgroups and other relatively simple tasks.

The injection of Sun's Mad Hatter desktop into the options mix now invites these organisations to go the next step -- replacing the Microsoft desktop suite with a comparable open source suite for those users for whom the software differences are judged to be relatively unimportant.

Notice that doing this will require:

1. The conversion of any corporate application clients needed by these users from Wintel to Lintel;
2. The conversion of the corporate desktop (for remote re-load and lockdown) from Wintel to Lintel; and,
3. Provision of appropriate help desk and other user support facilities for Lintel.

In reality the change from Wintel to Lintel is a bit more complicated than learning not to automatically reboot or reinstall, but most managers confronted with the Lintel opportunity are likely to believe themselves and their staff qualified to undertake it. Thus naive managers may be expected to see Lintel as something they can "slide in" with existing resources as a neat way to cut some costs while more qualified people will see the need for either outside help or significant re-training. In either case an attempted Lintel conversion is more likely to produce short term budget increases and dissatisfied users than immediate savings and a halo of corporate glory floating gently over the IT manager's head.

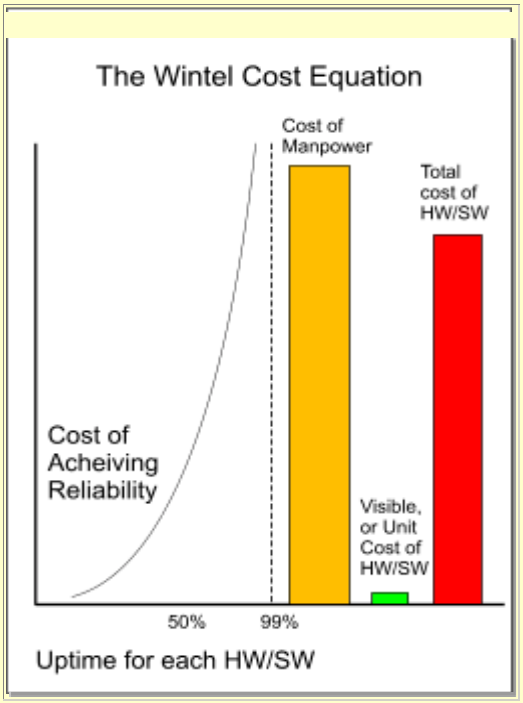
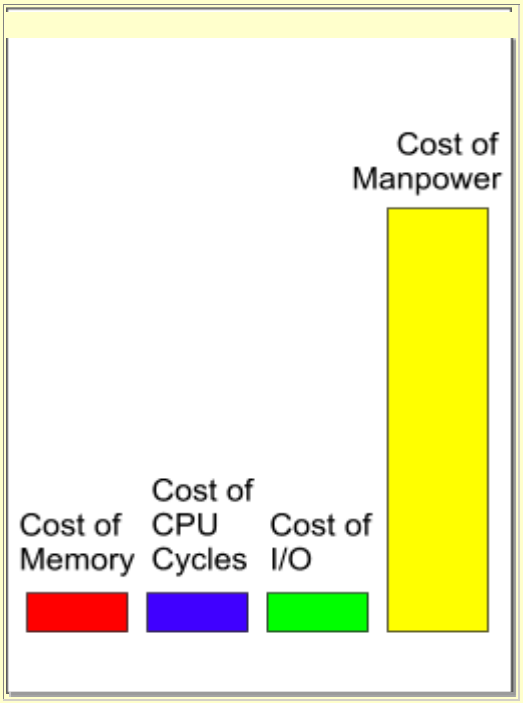
The underlying problems, of course, are the time lag on retraining and the need to perpetuate Wintel best practices like desktop lockdown, switched networks, and server centralization.

Give a bunch of MCSEs a bare bones six weeks off to get briefed on desktop Lintel, and you need to hire contractors to take their places. Try to wing it, and you'll have users rejecting the Lintel machines as unusable. Worse, when you eventually do get the Lintel help desk more or less functional and user despair down to a dull roar, you'll find yourself employing more MCSEs rather than fewer because the practices they implement have been continued into the Lintel environment --making things like the use of Active Directory even more complex and time consuming than before.

Use of the Unix Business Architecture eliminates all of these problems:

- First it removes the user empowerment lie fundamental to the client-server architecture by eliminating the pretence that the desktop is a personal, rather than corporate, resource and thus positions the IT group for an honest working relationship with users.
- Secondly it eliminates the need for most of the MCSE functions and thus the need to train, support, manage, and pay them.

  In a properly run UBA environment there is no help desk because there are no ambiguities about roles or failures. Systems delivers systems services: the responsibility is to deliver application processing as required. If something fails, that failure will be brutally obvious --the screen doesn't come on, the login fails, whatever it is the cause will be immediately obvious; no reboots, no network debugging, no re-installs, no ambiguity.

The Wintel Structure	Gave rise to these data center practices
	• Workgroup structure replicated in data center • Use of at least one physical server per application 'server" • Focus on fire fighting • Focus on help desk numbers • User separation and strong usage controls • Employee role separation • Strictly heirarchial decision making • Focus on managing and tracking roll-outs • Fixed nominal budgets plus ad hoc upgrade authorizations
The Unix Cost Structure	**Gave rise to these practices**
	• Focus on inter-personal communication and networking • Focus on real time business processing support • Focus on meeting user needs • Focus on system response time • Competence based decision making • Application integration through real-time data sharing • Employee empowerment and cross training • Hacking (adapting other people's code) and Open Source

From an applications perspective that means application help comes from the domain expert in the next cubicle, not a bored help desk clerk two continents away who tells the user to reboot.

From a management perspective it means that a good Unix sysadmin can run systems serving hundreds, not tens, of users. Sun, for example, is claiming that its ratio of support staff to users on just over 25,000 smart displays worldwide is currently running at better than 1:1000 and is headed for 1:2000 --compared to the Wintel average of about 1:30.

- Thirdly it eliminates product churn as a productivity detractor. Most smart displays, especially if over-configured to begin with, last five to seven years without replacement or failure. Users therefore typically get high powered, big screen, graphics with near absolute reliability and no noise or significant heat production while Systems can change servers, and software, with little direct impact on the user. Thus application upgrades affect users, but all the technology around those applications stays the same and the user is isolated from the techno-shock effects of having to learn whole new OS, E-communication, and applications suites every year or so.
- Fourthly, the role and positioning of the Systems organization changes dramatically. With client-server, as in the mainframe systems it emulates, control has to be centralized. Systems management cannot afford to allow a repeat of the chaos marking early client-server implementations as user groups fought to exert control and maintain the right to do their own thing with their own computers.

  With the UBA, however, control should be decentralized even as processing is centralized. Fundamentally, the cost implications of giving users more control about what they run, what defaults they use, and how they apply the system are minor while the corporate productivity benefits achievable through user directed Systems activities are substantial. This, of course, reverses the standard conceptualization in which Systems management acts as custodian of scarce corporate resources but that's one of the things Unix, and thus the UBA, does: it reverses the fundamental cost equation governing Systems management.

So how do the costs work out in realistic scenarios? That's what Part Three, below, is about.

Part Three: Alice Skips the Scene

`At any rate I'll never go there again!' said Alice as she picked her way through the wood. `It's the stupidest tea-party I ever was at in all my life!'

The scenario involves an all Wintel corporate organization with a head office, two sales and support offices, and three manufacturing plants. There are a total of about 1,000 day shift computer users and the company has a total of about 1,800 employees.

For the purposes of this comparison the company is assumed to have a full Peoplesoft suite in place and working as its core enterprise application. Since Peoplesoft is largely client independent, this assumption allows us to focus entirely on the desktop. Notice that the Wintel budget focuses on a continuation of existing practice while the UBA budget envisages complete replacement of the existing infrastructure.

Here's what key pieces in that Wintel budget look like:

Committed Hardware and Software upgrades					
	Replacement Hardware	**Replacement Software**	**Units this budget year**	**Current Unit Cost**	**Budgetary estimate**
Desktops; (300 refresh, 40 new)	17-inch Dell GX270; 256MB, 40 GB hard disk, 2.8GHz P3	Windows/XP Professional Office/XP Professional;12 mnths Norton	340	$1,143 $528	$568,140
Laptops; (90 refresh, 60 new)	Dell M60; 15.2" TFT, 512MB, 60GB hard disk, 1.7GHz P3; Wi-lan capable; basic case	Windows/XP Professional Office/XP Professional; 12 mths Norton	150	$3,638 $528	$624,900
Printers; (60 refresh, 10 new)	HP Laserjet 2200D with USB connector, 1200DPI		70	$489	$34,230
Servers; (26 refresh, two racks of 7 new)	Dell 2650, 2 x 2.8GHz, 4GB, 5 x 73GB/15K RAID 0,1; 2 x Intel 1000Pro NICs, CD, rack ready	Windows Server 2003 Enterprise Edition with 25 Client Licenses - Application servers are additional	40	$10,427 $3,295	$541,680
New Application Server Licenses		MS Commerce server MS SQL-Server (25 CALS) MS Identity Integration Server 2003 Enterprise Edition MS Exchange server 2000 - enterprise with 25 CALS	2 2 1 5	$6,999 $11,049 $24,999 $6,999	$118,188
22 switches, 4 routers, 8 Wi-lan	Dell 8-port switches, Cisco routers, Dell Wi-lan access severs		22 4 8	$489 $11,200 $899	$61,848
Total					$1,830,798

Notes:

1. Dell pricing from dell.com on 09/07/03;
2. Microsoft pricing from microsoft.com/catalog/ on 09/07/03.

Data Center Staffing					
	Desktop Support	**Server Support**	**Network Specialists**	**Development and Trouble shooting**	**Management**
MCP Level	20	4	1	0	1
MSSA Level	4	1	2	0	1
MCSE level	2	7	2	4	4
Average Salary	37,200	45,200	51,500	61,300	72,800
Total (includes 25% benefits and overheads)	$1,209,000	$678,000	$321,875	$306,500	$546,000
Total					$3,061,375

Notes:

1. Three budgeted MCP positions are actually in secretarial support
2. Estimates are approximated from the Computer world 16th annual salary survey

This year the current 30 month evergreen policy demands about a quarter of the budget, about 10% is for service expansion, and most of the remainder, amounting to about 60% of the total, goes directly to manpower.

Since Linux will run effectively on less hardware than the latest Microsoft Windows brand OS product the Lintel conversion has the result of extending the lifetime of gear in place. A CIO looking at desktop Lintel can choose between two ways of recognizing the resulting hardware savings:

1. He could immediately change the evergreen policy to a 48 month cycle instead of the typical 30 month cycle, load Linux on gear that would have been replaced under the 30 month cycle, and re-allocate existing gear to maximize usability among the remaining Wintel users.

   In the scenario used here, this approach produces annual budgetary savings (assuming Lintel runs around $95 per machine per year) of about $200,000 in licensing and $385,000 in hardware.

2. Or, he could cancel this year's upgrade cycle entirely in favor of budgeting to replace roughly half the installed desktop gear at next year's prices.

   This approach yields the same software saving but pulls much of next year's hardware savings into the current budget year.

Option one produces the higher discounted future value but no sensible CIO would prefer it because these savings would be reflected in long term cuts to the IT budget. Option two, in contrast, gives him immediate cash to affect the change with minimal long term budgetary impact.

In the scenario the potential immediate savings are substantial -- amounting to about $200,000 in software and $740,000 in hardware. Assuming that roughly half of this is returned to the corporate treasury as part of the negotiations leading up to executive committee approval for the change, he'll still have about $450,000 in this year's budget from which to pay for the transition.

The Rules in Monopoly 6.0
Organizations blessed with Enterprise 6.0 licensing agreements need to consider the trade-offs, if they have to make them, between abrogating the agreement and paying for a Microsoft OS for each Lintel machine.

Unfortunately an IT department can't just throw Linux on some user PCs and ride off into the sunset feeling ennobled by the experience; there are a few gotchas in the process that have to accounted for and all of them spell the same phrase: "manpower crisis." For example:

1. The knowledge and server infrastructures in place to support those new Linux devices is Wintel.

   What this means is first that the complexity of linking client devices to servers goes up tremendously when Lintel desktops are added to the mix and secondly that a conversion to Linux services in areas such as identity management, file and print support, and database hosting eventually needs to be staffed, started, and managed.

2. Users who are barely confident about their abilities to deal with an OS for which they get lots of external social support aren't likely to be all that excited about suddenly facing that dreaded enemy of progress: Unix; even if it is referred to by the politically correct name of Linux. Qualified, enthused, technical support, and lots of it, will be needed to get them over the hump.

   Worse, any significant desktop conversion will run into committed opposition from full time Windows hobbyists deeply embedded in the user community.

   Every major user group has at least one of these, and they're not on the IT payroll. They're the people with the PC magazines (and personal Wi-Fi access points) on their desks, the go-to people for local PC problems, the people users send to IT related meetings and to whom they defer when spreadsheets get corrupted or Word files magically disappear --and they're guaranteed to fight any change that devalues their expertise or threatens to send them back to work.

3. The older gear that was scheduled to just miraculously leave the building as part of the replacement roll-out now has to be cleaned, tested, reloaded, and re-allocated by people on the IT payroll.

The Peoplesoft Defense

For a small company with one or two technical people a Lintel conversion starting with servers and spreading to desktops can be very effective. In this situation the technical people work very closely with users and share their growing enthusiasm for, and expertise with, Linux *before* implementing it for user services.

In bigger organizations that kind of interaction isn't practical because the data center's internal size and external control issues mitigate against developing the kind of trust relationships they have to have with the user community for the transition to work.

The Peoplesoft defense against Oracle appears to have been strikingly effective --and original. By promising customers a full refund plus penalties if Oracle succeeds in taking them over, they built customer confidence, precipitated orders, and raised enormous cost barriers for Oracle.

Going Lintel on some client-server desktops offers bigger MCSE shops the same defense against Unix simply by doing what comes naturally --doing the best they can by applying what they know about Wintel to Lintel (without any implication of malfeasance or destructive intent) will destroy the user's expectation of success, make Wintel seem increasingly attractive, and create significant cost barriers to subsequent change.

All of these challenges can be met, but all require significant expertise and commitment that this organization cannot possibly have given its all-Wintel infrastructure. A sensible CIO would therefore step back from this brink and build that expertise through server conversion before proceeding with desktop change.

The immediate savings from server conversion aren't as large, perhaps only $200,000 in the first year, but the opportunity to build confidence and expertise swamps the cost difference and gives the eventual Lintel desktop conversion a much higher chance of success. In this case, for example, the CIO could cancel most of his new server development and invest those monies in building Unix experience by implementing identity services, single logon, and departmental file and print sharing using Linux.

With those pieces in place, the downstream Lintel desktop conversion wouldn't need to attach those machines to a Wintel server infrastructure and thus be both simpler and far less likely to run afoul of something like a Microsoft service pack that breaks Lintel connectivity. More importantly, however, the on site experience needed to implement the server end successfully would eventually make the desktop conversion both safer and easier.

The question, however, is where that expertise is going to come from. There are three non exclusive choices:

1. Outsource the problem to a company, like IBM, which promises the required expertise.
2. Build on the expertise available internally by encouraging experimentation and retraining while using consultants to help get over the more difficult spots.
3. Hire some Unix people and support their efforts to change the way things are done in the IT group.

Of these the outsourcing solution is usually the easiest to sell to top management; after all they've been baffled for years at the CIO's failure to deliver cheap, efficient, reliable Wintel computing.

Option two is the most common. Unfortunately it has the downside that MCSEs encouraged to learn Linux will generally learn to apply Linux in ways which mimic the Windows brand products they're more familiar with. The net effect, of course, is usually to turn Linux into a second rate Wintel emulation with consequent costs combining a corporate failure to benefit from the change with increased operational complexity in IT.

For example, Microsoft recently published a comparison of Linux and Windows Server 2003 for file and print services. In this comparison the people who did the tests, Veritest (formerly zdlabs), put eight network cards into an 8 CPU Linux machine and used the processor affinity capabilities in Red Hat Advanced Server 2.1 to tie each card to a processor. This is standard practice in the Wintel environment but ignores the existence and value of Unix SMP to unnecessarily add administrative complexity while raising costs and reducing throughput.

Option three is the most difficult, but also the only one that offers a realistic chance at long term success. At its best this process, which I refer to as data center defenestration, requires an extremely rare combination of strong leadership and knowledge but leads to the kind of low cost, high reliability computing discussed below. At its worst, this puts MCSE or mainframe qualified people in the position of having to hire, evaluate, and support employees whose Unix expertise gives them opinions and attitudes that make them pariahs in the Wintel or mainframe world their managers come from.

The worst problem about this is that there's a kind of catch-22 aspect to it. Hire people who are genuinely qualified, and they'll start looking for another job just as soon as they find themselves being asked to implement a Lintel based client-server architecture --because that's fundamentally a Wintel thing and inappropriate in Unix. Get people whose knowledge of Unix is sufficiently shallow, and they'll work on the help desk alongside the MCPs but their lack of real depth will cripple the results the company gets and leave Lintel every bit as frustrating a user environment as Wintel.

The Unix Business Architecture introduced earlier provides a nice answer to this set of problems mainly because it allows the company to eliminate the client-server architecture and, with it, most of the IT people.

From a structural perspective the UBA as implemented for the scenario company would consist of two data center teams separately responsible for the two halves of a mirrored system built around a pair of Sun Fire 6800s delivering both application and Mad Hatter office automation services to Sun Ray desktops and Mac laptops.

Unix Architecture Components and Cost					
Cost Source	**Type**	**Configuration**	**Unit Cost and discounts**	**Units**	**Capital or Annual Cost**
Primary servers	Sun Fire 6800, 24 x 1.2GHz CPUs, 96GB RAM, 4 x 73GB	Company wide Peoplesoft/Sybase; head office Mad Hatter operations; and company wide backup/restore.	$842,895 x 0.85	2	$1,433,074
Plant Site Support	Sun Fire V1280, 12 x 1.05Ghz CPUs, 24GB RAM, 2 x 73GB	Mad Hatter and MacOS X Support	$141,995 x 0.85	3	$362,087
Sales Office Support	Sun Fire V880, 6 x 1.05Ghz CPUs, 32GB RAM, 2 x 73GB	Mad Hatter and MacOS X Support	$109,995 x 0.85	2	$186,991
Disk Storage	3320 SCSI Arrays/Controllers - 12 x 73GB, 15K RPM[5]	All set up as RAID 1 plus mirroring to paired device; 2 on each V880; 4 on each V1280; 8 on each 6800	$20,995 x 0.6	32	$403,104
Desktops	19" Sun Ray	All fixed desktops	$799[5]	1000	$799,000
Laptops	15" TFT Mac G4; 512MB, 1GHz, 60GB disks, Airport, SuperDrive	Sales support	$2,599	150	$389,850
Printers	HP or Kyocera	16 and 24PPM mixed BW/Color full Postscript level III	$3,000	30	$90,000
Cell Phones	Motorola (Linux based)	Executive and sales support	$369 +$39.95/mth	120	$146,088
Networking	Avaya S8300 (Linux) with P330/G600	Includes telecom handsets, PBX, VOIP, TCP/IP routing, VPN and re-cabling	Averages $840/usr	1000	$840,000
Systems Staff	Unix and operational skill-sets	Grouped as two independent, 7 person, teams	$72,000	14	$1,209,600[6]
Office of CIO	CIO and two administrative assistants	(Excludes facility costs)	$105K $40K	1	$222,000
Five Year annualised cost					$2,934,660

[1]Notes:

1. Totals may appear incorrect due to rounding
2. Pricing is from Sun, Avaya, Apple, websites on 09/07/03
3. Server totals include 20% annual maintenance
4. Playphone capital cost multiplied by 2 to account for product evolution
5. The 19" Wanray, 3320 15K/RPM RAID set, and CPU/Disk upgrades on the V1280 are all expected to be announced shortly.
6. Salary totals include 20% overheads and benefits
7. Revenue applications licensing ignored for both cases
8. The Avaya telecom gear includes everything needed - from handsets to 100baseT connections.

Data center staffing will consist of:

1. Four sysadmins in each group;
2. Two application specialists in each group;
3. One administrative support person in each group; and,
4. Three people in the office of the CIO - two administrative, one management.

for a total of 17 people split into two operational groups both reporting to the CIO.

Notice that the five year total, $14,673,303, includes the costs associated with the mobile phones and the addition of IP integrated voice and video conferencing to every desktop on the network. It is, nevertheless, 40% less than the five year total of the Wintel costs shown.

Cost savings like this are important but not as important as other benefits. For example:

1. By design and implementation, this Unix architecture would be functionally impervious to the security and reliability issues that can turn the Windows architecture solution into a daily struggle.
2. IT staffing is enormously simplified. With somewhere between a quarter and a third as many positions to fill, companies can pick better people, offer higher rewards, and build stronger, more stable, operational teams.

   Initially, of course, it will not usually be possible to hire 14 good Unix people during a reasonably short recruitment period but the transition doesn't need that many. Companies starting with a good CIO and two experienced Unix team leaders can usually flesh out the rest of the staffing needed by hiring science graduates with little or no previous work experience while relying on Sun to do initial setup. Given the right atmosphere and leadership, such people become solid sysadmins and DBAs much more quickly than IT professionals who must first unlearn inappropriate habits of thought and action.

3. Overall system design and operation is tremendously simplified too. There are more software choices, but fewer opportunities to make the underlying architecture more complex than it needs to be. As a result the typical implementation is likely to be closer to the ideal than the average Wintel client-server implementation gets and, of course, many of the finger pointing opportunities associated with Wintel organizational complexity simply go away with the smaller, more empowered, Unix staff.
4. The structure essentially guarantees processing continuity under all but the most extreme conditions. Fundamentally everything, most importantly including the sysadmin teams --because almost all failures involving high availability gear like the Sun Fire 6800 start with administrative error-- is duplicated. Users might be unhappy about system response during an outage, but nothing comes to a complete halt and no data would be lost even if one of the data centers blew itself up.
5. System usage is enormously simplified for users. To use a smart display, you turn it on and log in. With Sun Rays, in particular, you pop in your smart card and get your home system --regardless of whether you are in your office, a boardroom, or visiting a plant 3,000 miles from home.

   From the key application delivery perspective the reliability of the system means that there is no ambiguity about problems. If a machine fails, it fails, there's no reboot, no debugging, nothing of that kind occurs. Instead, the user just finishes whatever he or she was working on using someone else's display until a replacement can be brought over and plugged in --no setup required. As a result there is no role for a help desk and users get application assistance from the domain experts they work with every day, not someone whose first concern is determining whether the user's problem lies with the user, the desktop PC, the network, or the server.

Those kinds of benefits are not easily quantified, but make it possible for the CIO in this scenario to sell both the vision and the reality to top executives simply by hiring one Unix expert to set up a small scale pilot completely outside normal day to day data center operations. Once that's approved, of course, he can recruit a mere handful of people and proceed without regard to the incompatibility between the MCSE mindset and Unix.